REGISTERATION FORM

### Overview

This task involves creating a GUI registration form using the Tkinter library. The form includes:

* Name
* Email
* Age
* Phone Number
* Village
* Pincode
* Country
* State
* Father’s Name
* Mother’s Name
* Father’s Number
* Mother’s Number
* Gender (with radio buttons)
* Submit button

### Prerequisites

* Python installed on your system.
* Tkinter library installation code : pip install tkinter

### Implementation Steps

1. **Import Required Libraries:**

import tkinter as tk

from tkinter import messagebox

1. **Create the Submit Function:** This function will handle form submission and validation.

def submit():

name = name\_entry.get()

email = email\_entry.get()

age = age\_entry.get()

gender = gender\_var.get()

# Validation for empty fields

if name == "" or email == "" or age == "":

messagebox.showerror("Error", "Please fill in all fields")

else:

messagebox.showinfo("Success", f"Registration Successful!\nName: {name}\nEmail: {email}\nAge: {age}\nGender: {gender}")

1. **Create the Main Window:** Set up the main application window.

root = tk.Tk()

root.title("Registration Form")

1. **Add Labels and Entry Fields:** Define labels and entry fields for name, email, and age.

# Name

name\_label = tk.Label(root, text="Name:")

name\_label.grid(row=0, column=0, padx=10, pady=5)

name\_entry = tk.Entry(root)

name\_entry.grid(row=0, column=1, padx=10, pady=5)

# Email

email\_label = tk.Label(root, text="Email:")

email\_label.grid(row=1, column=0, padx=10, pady=5)

email\_entry = tk.Entry(root)

email\_entry.grid(row=1, column=1, padx=10, pady=5)

# Age

age\_label = tk.Label(root, text="Age:")

age\_label.grid(row=2, column=0, padx=10, pady=5)

age\_entry = tk.Entry(root)

age\_entry.grid(row=2, column=1, padx=10, pady=5)

1. **Add Gender Radio Buttons:** Use radio buttons to select gender.

gender\_label = tk.Label(root, text="Gender:")

gender\_label.grid(row=3, column=0, padx=10, pady=5)

gender\_var = tk.StringVar()

male\_radio = tk.Radiobutton(root, text="Male", variable=gender\_var, value="Male")

male\_radio.grid(row=3, column=1, padx=10, pady=5)

female\_radio = tk.Radiobutton(root, text="Female", variable=gender\_var, value="Female")

female\_radio.grid(row=3, column=2, padx=10, pady=5)

1. **Add Submit Button:** Create a button to submit the form.

submit\_button = tk.Button(root, text="Submit", command=submit)

submit\_button.grid(row=4, column=0, columnspan=2, padx=10, pady=10)

1. **Run the Main Event Loop:** Start the Tkinter event loop to run the application.

root.mainloop()

* OTHERS :
* Village , country ,state ,Father’s name and mother’s name same as name
* Pincode, Father’s Number and Mother’s Number same as phone number

### Complete Code

Here’s the complete code for the registration form:

import tkinter as tk

from tkinter import messagebox

# Define the submit\_form function

def submit\_form():

name = entry\_widgets[0].get()

email = entry\_widgets[1].get()

age = entry\_widgets[2].get()

phone\_number = entry\_widgets[3].get()

village = entry\_widgets[4].get()

pincode = entry\_widgets[5].get()

country = entry\_widgets[6].get()

state = entry\_widgets[7].get()

gender = gender\_var.get()

father\_name = entry\_widgets[8].get()

mother\_name = entry\_widgets[9].get()

father\_number = entry\_widgets[10].get()

mother\_number = entry\_widgets[11].get()

# Print or process the collected data as needed

print("Name:", name)

print("Email:", email)

print("Age:", age)

print("Phone Number:", phone\_number)

print("Village:", village)

print("Pincode:", pincode)

print("Country:", country)

print("State:", state)

print("Gender:", gender)

print("Father's Name:", father\_name)

print("Mother's Name:", mother\_name)

print("Father's Number:", father\_number)

print("Mother's Number:", mother\_number)

# Optional: Show a message box indicating successful submission

messagebox.showinfo("Success", "Registration Successful!")

# Create main window

root = tk.Tk()

root.title("Registration Form")

# Get screen width and height

screen\_width = root.winfo\_screenwidth()

screen\_height = root.winfo\_screenheight()

# Set the width and height of the window and calculate the x and y coordinates

window\_width = 400

window\_height = 600

x = (screen\_width // 2) - (window\_width // 2)

y = (screen\_height // 2) - (window\_height // 2)

# Set the window size and position

root.geometry(f"{window\_width}x{window\_height}+{x}+{y}")

# Create and place widgets

heading\_label = tk.Label(root, text="REGISTRATION FORM", font=("Arial", 16))

heading\_label.pack(pady=20)

fields\_frame = tk.Frame(root)

fields\_frame.pack()

# Entry fields

fields = ["Name", "Email", "Age", "Phone Number", "Village", "Pincode", "Country", "State", "Father's Name", "Mother's Name", "Father's Number", "Mother's Number"]

entry\_widgets = []

for i, field in enumerate(fields):

label = tk.Label(fields\_frame, text=field + ":")

label.grid(row=i, column=0, sticky="e", pady=5)

entry = tk.Entry(fields\_frame)

entry.grid(row=i, column=1, padx=10, pady=5)

entry\_widgets.append(entry)

# Gender radio buttons

gender\_var = tk.StringVar(value="Male")

gender\_label = tk.Label(fields\_frame, text="Gender:")

gender\_label.grid(row=len(fields), column=0, sticky="e", pady=5)

male\_radio = tk.Radiobutton(fields\_frame, text="Male", variable=gender\_var, value="Male")

male\_radio.grid(row=len(fields), column=1, padx=10, pady=5, sticky="w")

female\_radio = tk.Radiobutton(fields\_frame, text="Female", variable=gender\_var, value="Female")

female\_radio.grid(row=len(fields)+1, column=1, padx=10, pady=5, sticky="w")

# Submit button

submit\_button = tk.Button(root, text="Submit", command=submit\_form)

submit\_button.pack(pady=20)

root.mainloop()

### Explanation

* **Labels and Entry Fields:** Used to collect user input for name, email, and age.
* **Radio Buttons:** Allow the user to select their gender.
* **Submit Button:** Triggers the submit function, which validates input and displays a success message.
* **Validation:** Checks if any fields are empty and displays an error message if necessary.
* **Layout:** Uses a grid layout to organize the form elements neatly.

### Conclusion

This guide covers the basics of creating a simple registration form using Tkinter. You can enhance this form by adding more fields, validation, or saving the data to a file or database. This task provides a solid foundation in GUI development with Python and Tkinter.